


# Guide to Resources for CSAI202

A list of useful resources for learning razor pages, ordered by personal preference and organized into categories. Feel free to suggest more resources by creating a PR.

Content:

- [Full Resources](#)
- [Crash Courses and Cheatsheets](#)

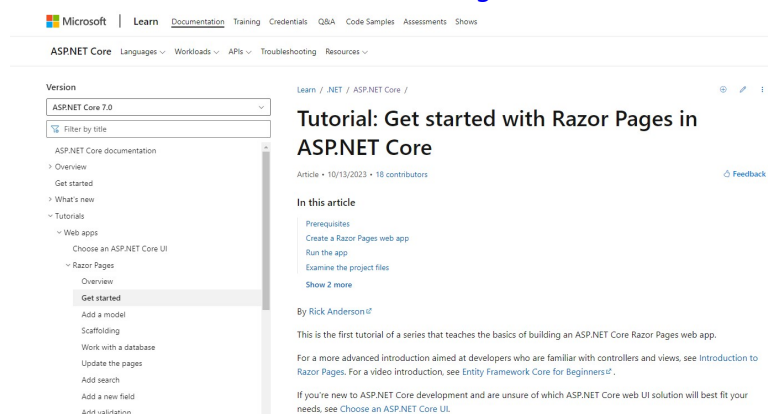
## Full Resources

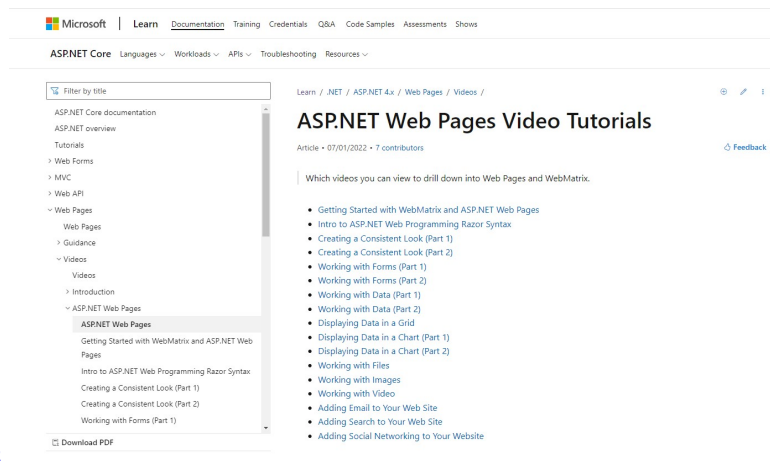
1.  [LearnRazorPages.com](#) A very good documentation and explanation website for razor pages



concepts and practices.

2.  [Tutorial: Get started with Razor Pages in ASP.NET Core - from Microsoft](#)

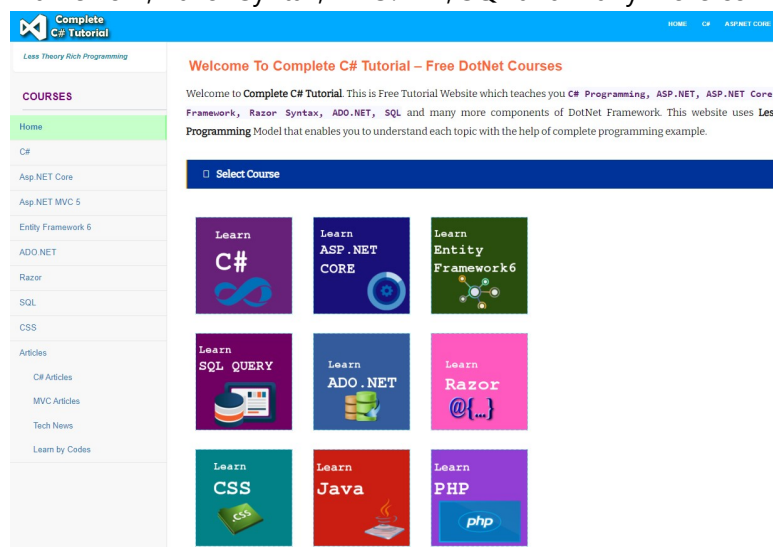




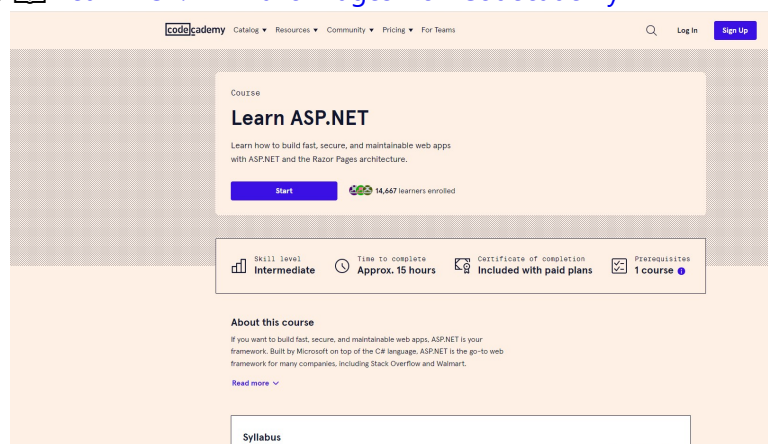
### 3. 📺 Microsoft Video Tutorials

### 4. 📖 Complete C# Tutorial Website

A Free Tutorial Website that teaches you C# Programming, ASP.NET, ASP.NET Core, MVC, Entity Framework, Razor Syntax, ADO.NET, SQL and many more components of DotNet Framework.



### 5. 📖 Learn ASP.NET Razor Pages - on Codecademy



### 6. 📺 CodeWithGopi, ADO.NET Playlist, 12 videos

### 7. 📺 Questpond, Session and ViewState in ASP.NET

# Cheatsheets and Crash Courses

### Bullet Points:

- C# code is case sensitive
- Razor comment: `@* your comment *@`
- All new razor pages must be added inside the *Pages* folder
- *\_Layout.cshtml* is a partial view which encapsulates all razor pages inside the *Pages* folder
- Handler methods must be public. They can return either `void` or `ActionResult`
- An html form must have a **method** attribute, and it must be equal to **post**.  
`<form method="post">`
- Add an **asp-page-handler** attribute to the form if there are multiple `OnPost()` methods in one page
- To inject the DB class -> at program.cs file add

1. Add namespace:  
**using solutionName.Models;**
2. **builder.Services.AddSingleton<DB>();**

### Tag Helpers:

1. `asp-route-id`
2. `asp-page-handler`
3. `asp-for`
4. `asp-page`
5. `asp-validation-for`

### Structure of handler methods:

- Handler methods must use the following naming convention:  
**On + HTTP verb + Handler Name + (Parameters)**

used for the request

Optional

Optional

```
public void OnGet(){}
```

### ADO.net

Connecting web app with database using ADO.net:

1. Include necessary packages using `System.Data;` and `System.Data.SqlClient;`
2. open a connection `SqlConnection`  
`string constr = ""`  
`SqlConnection con = new SqlConnection(constr);`  
`con.Open();`
3. Form a query  
`string query = ""`
4. Form a **SqlCommand (cmd)**  
`SqlCommand cmd = new SqlCommand(query, con);`
5. Execute the command
  - a. `cmd.ExecuteReader() ->` (Returns table)
  - b. `cmd.ExecuteNonQuery() ->` (Returns 1 row affected)
  - c. `cmd.ExecuteScalar() ->` (Returns single value)
6. Catch any exceptions (optional)
7. Close the connection  
`con.Close();`

### DataTable

- needed namespace:  
**using System.Data**
- Property:  
**public DataTable dt { get; set; }**
- Object:  
**DataTable dt = new DataTable();**

### Take Input From the User

(To pass parameters from the frontend to the backend)

### Frontend:

## 1. CIE206 SP23 CheatSheet

codecademy

Catalog ▾ Resources ▾ Community ▾ Pricing ▾ For Teams

Log In

Sign Up

Cheatsheets / Learn ASP.NET

ASP.NET: Razor Syntax

Print Cheatsheet

Share

TOPICS

ASP.NET: Razor Syntax

ASP.NET: Page Models

ASP.NET: Databases

ASP.NET: Middleware

ASP.NET: Dependency Injection

### Razor Syntax

Razor Syntax allows you to embed code (C#) into page views through the use of a few keywords (such as `@`), and then have the C# code be processed and converted at runtime to HTML. In other words, rather than coding static HTML syntax in the page view, a user can code in the view in C# and have the Razor engine convert the C# code into HTML at runtime, creating a dynamically generated HTML web page.

```
@page
@model IndexModel
<h2>Welcome</h2>

<ul>
  @for (int i = 0; i < 3; i++)
  {
    <li>@i</li>
  }
</ul>
```

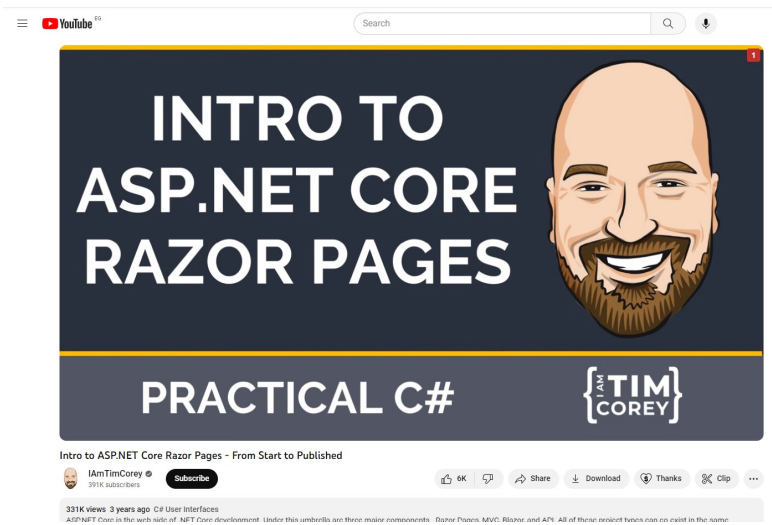
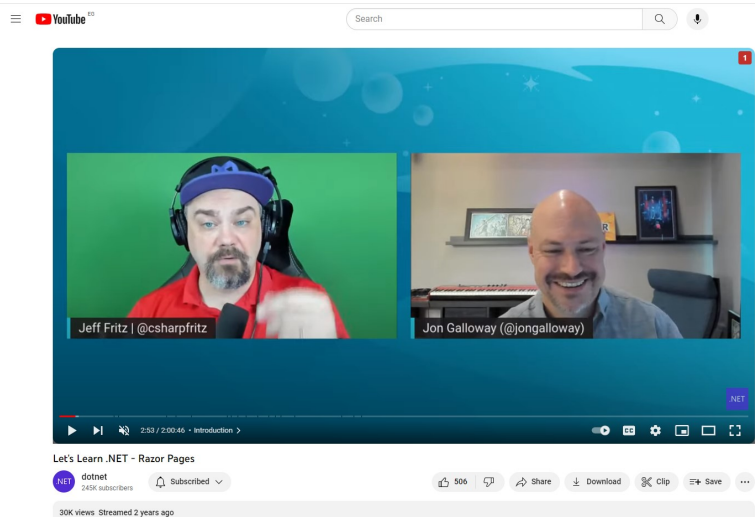
### The @page Directive

In a Razor view page (.cshtml), the `@page` directive indicates that the file is a Razor Page.

```
@page
@model IndexModel
```

## 2. Codecademy Cheatsheet

3. Let's Learn .NET Razor Pages - by dotnet on Youtube



4. Intro to Razor Pages - Tim Corey